

Init3DCurveC.txt

功能：求插值三维曲线数据点的幂基参数多项式系数

输入参数：DataPointx,DataPointy,DataPointz-依次是数据点的x,y,z分量。

输入参数：m\_Cx,m\_Cy,m\_Cz-依次是幂基参数多项式系数矢量的x,y,z分量。

```
Void Init3DCurveC(int n, CArray<double, double> &UV,
CArray<double, double> &DataPointx,
CArray<double, double> &DataPointy,
CArray<double, double> &DataPointz,
CArray<double, double> &Cx,
CArray<double, double> &Cy,
CArray<double, double> &Cz)
{
    CArray<CArray<double, double>, CArray<double, double>&> AA;
    CArray<double, double> tx, ty, tz;
    UV.SetSize(n);
    for(int i=0; i<n; i++)
    {
        tx.Add(double(DataPointx[i]));
        ty.Add(double(DataPointy[i]));
        tz.Add(double(DataPointz[i]));
    }
    AA.SetSize(n);
    for(i=0; i<n; i++) AA[i].SetSize(n);
    Cx.SetSize(n);
    Cy.SetSize(n);
    Cz.SetSize(n);
    for(i=0; i<n; i++)
    {
        double temp;
        for(int j=0; j<n; j++)
        {
            temp=1.0;
            for(int l=0; l<j; l++) temp=temp*UV[i];
            AA[i][j]=temp;
        }
    }
    int chl=0;
    for(int tt=0; tt<n-1; tt++)
    {
        double d=AA[tt][tt];
        bool n_move=false;
        for(int j=tt+1; j<n; j++)
        {
            if(fabs(d)<fabs(AA[j][tt]))
            {
                d=AA[j][tt];
                chl=j;
                n_move=true;
            }
        }
        double tem1, tem2, tmpx, tmpy, tmpz;
        if(n_move)
        {
            for(int mm=tt; mm<n; mm++)
            {
                tem1=AA[tt][mm];
                tem2=AA[chl][mm];
                AA[tt][mm]=tem2;
                AA[chl][mm]=tem1;
            }
            tmpx=tx[tt], tmpy=ty[tt], tmpz=tz[tt];
            tx[tt]=tx[chl], ty[tt]=ty[chl], tz[tt]=tz[chl];
            tx[chl]=tmpx, ty[chl]=tmpy, tz[chl]=tmpz;
        }
        for(j=tt+1; j<n; j++)
        {
            double l=AA[j][tt]/AA[tt][tt];
            tx[j]=tx[j]-l*tx[tt];
            ty[j]=ty[j]-l*ty[tt];
            tz[j]=tz[j]-l*tz[tt];
            for(int v=0; v<n; v++)
                AA[j][v]=AA[j][v]-l*AA[tt][v];
        }
    }
    Cx[n-1]=tx[n-1]/AA[n-1][n-1];
    Cy[n-1]=ty[n-1]/AA[n-1][n-1];
}
```

```

Cz[n-1]=tz[n-1]/AA[n-1][n-1];
for(int ll=n-2;ll>=0;ll--)
{
    double xx=0.,yy=0.,zz=0.;
    for(int k=ll+1;k<n;k++)
    {
        xx=xx+AA[ll][k]*Cx[k];
        yy=yy+AA[ll][k]*Cy[k];
        zz=zz+AA[ll][k]*Cz[k];
    }
    Cx[ll]=(tx[ll]-xx)/AA[ll][ll];
    Cy[ll]=(ty[ll]-yy)/AA[ll][ll];
    Cz[ll]=(tz[ll]-zz)/AA[ll][ll];
}
}

```